

App. No. 09/997,057

Amendment Dated: July 24, 2006

Reply to Final Office Action of April 24, 2006

REMARKS/ARGUMENTS

The claims have been amended as set forth above and remain in this application for further consideration. No new matter has been added. Applicants believe the claims are allowable over the cited references.

I. Rejection of Claims 1 - 20 Under 35 U.S.C. §101

Claims 1 through 20 are rejected under 35 U.S.C. §101 because it is believed that the claim language fails to identify a useful result that allows the functionality of the claims to be realized. Applicants respectfully disagree.

In making this determination, the Office Action cites *In re Warmerdam* (hereinafter "Warmerdam"). 33 F.3d 1354, 31 USPQ2d 1754 (Fed. Cir. 1994).

Warmerdam stands for the proposition that abstract ideas or laws of nature, which constitute descriptive material, are non-statutory. *See Warmerdam*, at 31 USPQ2d at 1759. However, when functional descriptive material is recorded on some computer-readable medium *it becomes structurally and functionally interrelated to the medium* and will be statutory in most cases *since use of technology permits the function of the descriptive material to be realized*. *See In re Lowry*, at 32 F.3d 1579, 1583-84, 32 USPQ2d 1031, 1035 (Fed. Cir. 1994) (stating that a claim to a data structure stored on a computer-readable medium that increases computer efficiency is statutory), *and Warmerdam*, at 33 F.3d at 1360-61, 31 USPQ2d at 1759 (stating that a claim to a computer having a specific data structure stored in memory is a statutory product-by-process claim). (Emphasis added). In light of the above case law, applicant asserts that the claims are allowable under 35 U.S.C. § 101.

The specification specifically recites useful results. The specification recites as follows:

"Namespace hierarchies in accordance with the present invention provide many advantages over other namespace designs. For example, the technique of the present invention provides a convenient way to determine the source of a name. If the namespace associated with the source of the name is unknown, that namespace may be conveniently retrieved and studied. In contrast, in C for example, namespaces are included into a flat naming universe that does not provide any indication of the source of a name. This makes reading code very difficult.

App. No. 09/997,057

Amendment Dated: July 24, 2006

Reply to Final Office Action of April 24, 2006

In another example, the naming technique of the present invention allows multiple namespace hierarchies to be created rather than enforcing one hierarchy. The present invention allows grouping names into namespaces and then assigning a unique identifier to that grouping. For instance, returning briefly to FIGURE 2, the Accounts Receivable namespace 203 is part of at least two namespace hierarchies—a hierarchy that includes the Business Management namespace 222 and the Accounting namespace 220, and a hierarchy that includes only the Collections namespace 221. Accordingly, depending on the developer's needs, the objects declared in the Accounts Receivable namespace 203 may be accessed through either hierarchy. Namespace designs that require agreement on one "right" hierarchy demand an almost unachievable goal. In contrast, the technique of the present invention allows source units to be moved without affecting the resolution of external references. A namespace may be defined for one source unit and changed very easily by redirecting the references from one subtree to another.

In yet another example, the naming technique of the present invention provides a convenient and unobtrusive migration path from legacy or standard-named namespaces, such as those currently defined by C++, to a namespace hierarchy in accordance with the present invention. When converting from legacy namespaces, a GUID is added to each standard named namespace to generate a root namespace in accordance with the present invention. This may be done at any time. Even after the "old" standard named namespace has been converted, the "old" namespace may remain available for use in the "old" manner as long as needed. Thus, the present invention does not interfere with existing implementations.

In still another example, the migration of standard or legacy namespaces to namespace hierarchies in accordance with the present invention may involve "helper namespaces" that import the newly defined namespaces and re-export their leaf definitions in a manner that reflects the reorganization of the old namespaces. Then, any existing source code may be redirected to the new "helper namespaces" by importing the new "helper namespaces" and referencing the associated GUID. In addition, the existing source code may later be modified to remove the "helper namespaces". The namespace hierarchies may also be conveniently modified to accommodate semantic changes to leaf definitions and reorganization of the namespace hierarchy.

In still another example, the import/export mechanism of the present invention allows the creation of arbitrary parallel namespace hierarchies that reflect multiple taxonomies. The imports may use local names to resolve any conflicts that may arise from collisions with recommended names. At any point of reference, the namespace hierarchies are anchored in a root GUID. It is desirable for the root GUID to be expanded to easily and conveniently illustrate

App. No. 09/997,057

Amendment Dated: July 24, 2006

Reply to Final Office Action of April 24, 2006

the corresponding definition for the namespace. In one embodiment, it is envisioned that the GUID may be stored in a commonly accessible location, such as a system registry. The development tools for editing, debugging and browsing may then reference the GUID in the commonly accessible location to obtain its definition.

In still another example, because the namespace hierarchy has an associated GUID, the namespace is uniquely identified. In one implementation, it is envisioned that namespaces will not be removed from another namespace hierarchy once the namespace has been published. However, new names may be added to the published namespace. Existing names will not be removed. The use of a name may be discontinued, or in other words deprecated. Once the name is discontinued, it may not be revived.

In yet another example, namespace hierarchies may be copied to make their definitions widely available. However, it is envisioned that there is one master copy of the namespace that is controlled by some authority. This authority is responsible for maintaining the master copy. The duration of the namespace may be indefinite or may have a defined expiration date or a defined non-use date. If the contract has a finite lifetime, the authority may be responsible for renegotiating or renewing the lifetime of the namespace." *Specification*, at Page 9, line 25 through Page 11, line 26.

Accordingly, Applicants assert that claims 1-20 are allowable under 35 U.S.C. §101.

II. Rejection of Claims 1 - 20 Under 35 U.S.C. §112, Second Paragraph

Claims 1 through 20 are rejected under 35 U.S.C. §112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which the Applicants regard as the invention. The Office Action asserts that the claim language is unclear as to what is being pointed out by the "locally modifiable, name portion" and the "globally unique identifier portion". The claims have been amended as set forth above and Applicants believe that the 35 U.S.C. §112, second paragraph, rejection is obviated.

III. Rejection of Claims 1 - 20 Under 35 U.S.C. §102(e)

Claims 1 - 20 are rejected under 35 U.S.C. §102(e) as being anticipated by U.S. Patent 6,611,844 issued to Saulpaugh, et al. (hereinafter referred to "Saulpaugh"). Applicants respectfully disagree with the rejection. Independent Claim 1 has been amended to recite the following elements not taught or otherwise suggested by Saulpaugh.

App. No. 09/997,057

Amendment Dated: July 24, 2006

Reply to Final Office Action of April 24, 2006

"...a first definition data field defining the first data structure as a first namespace, the first definition data field including a locally modifiable common name portion and a global unique identifier portion, wherein the common name portion *is modifiable to provide localized naming of the first namespace when the first namespace is imported into a second namespace, wherein the unique identifier portion is maintained to identify the first namespace when importing the first namespace into the second namespace*, wherein the common name portion locally identifies the first namespace in a human-readable manner, the unique identifier portion globally distinguishes the first namespace from the second namespace to allow the first namespace to be imported into the second namespace *without a conflict associated with the common name portion of the first namespace and a common name portion of a second namespace.*"

Applicants cannot find any such teaching or suggestion in Saulpaugh. Saulpaugh teaches a method of database storage. The portion cited in the Office Action recites as follows:

"FIG. 4 illustrates the hierarchical nature of the JSD. In one embodiment, the JSD is divided into six standard namespaces, or sub-trees of related entries, which are created when JavaOS starts: Temp, Device, Interface, Alias, Software, and Config. Entries within a given namespace share common characteristics. A default namespace manager manages each namespace, controlling how entries are created, added, accessed, removed, and updated for a particular namespace. When an entry is published (that is, added to the database and thus made public), it inherits its parent's namespace manager by default." *Saulpaugh*, at column 10, line 66 - column 11, line 9.

Applicants can find no teaching of the combination of elements recited in claim 1 in this portion of Saulpaugh. Also, Applicants cannot find any teaching in Saulpaugh in its entirety of the combination of elements set forth in independent claim 1. With regard to independent claims 8, 13 and 18, Applicants maintain the same arguments as set forth above in support for independent claim 1. Claims 2, 7, 9-12, 14-17, and 19-20 depend from independent claims 1, 8, 13, and 18, respectively. Applicants believe the dependent claims are allowable for at least the same reasons set forth above for the independent claims.

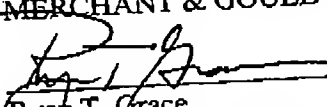
App. No. 09/997,057
Amendment Dated: July 24, 2006
Reply to Final Office Action of April 24, 2006

IV. Request For Reconsideration

In view of the foregoing amendments and remarks, all pending claims are believed to be allowable and the application is in condition for allowance. Therefore, a Notice of Allowance is respectfully requested. Should the Examiner have any further issues regarding this application, the Examiner is requested to contact the undersigned attorney for the applicant at the telephone number provided below.

Respectfully submitted,

MERCHANT & GOULD P.C.


Ryan T. Grace
Registration No. 52,956
Direct Dial: 206.342.6258

MERCHANT & GOULD P.C.
P. O. Box 2903
Minneapolis, Minnesota 55402-0903
206.342.6200

